DATA SCIENCE
MEI/1

University of Beira Interior

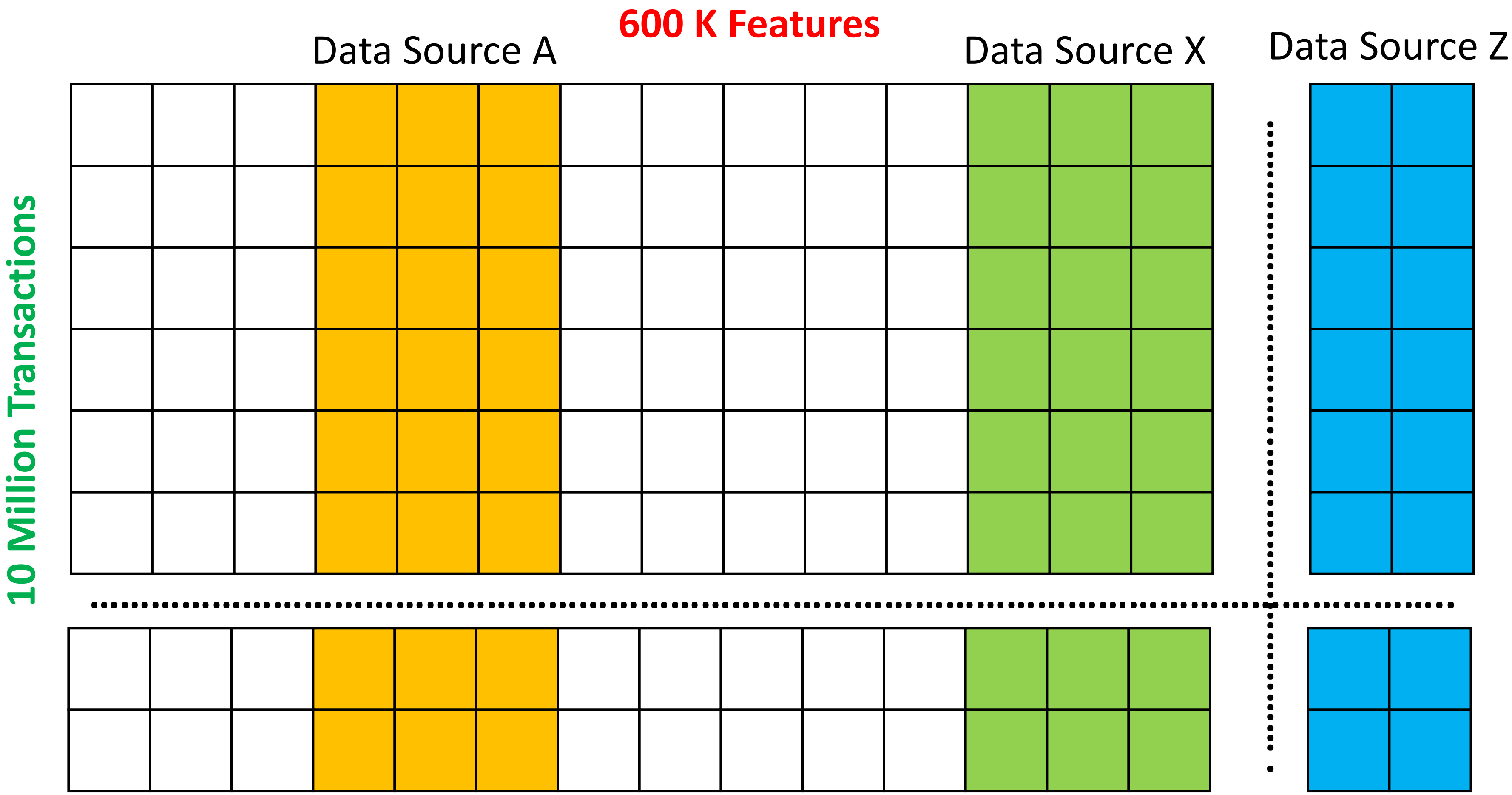
Department of Informatics

Hugo Pedro Proença,

hugomcp@di.ubi.pt, **2021/22**

ETL Output

- Suppose that the output provided by an **ETL** data transformaton process yielded the following data source:



ETL Output: Intractable Outputs

- Even considering that each elemento of the data set was stored in “***int32***” format
  - It is an immutable value type that represents signed integers with values that range from negative 2,147,483,648 through positive 2,147,483,647
  - Even this relativelly small data type would yield 10.000.000 x 600.000 x 32 bits:
    - 19 200 000 000 000 bits
    - 2 400 000 000 000 bytes
    - approx. 2 400 000 000 KB
    - approx. 2 400 000 MB
    - approx. 2 400 GB
    - approx. 2.4 TB
- This kind of data could easily become intractable, from the computational perspective.

Statistics

- The **Covariance Matrix** contains all covariance values between every possible dimension of a feature space :

$$C^{n\times n} = (c_{i,j},\ c_{i,j} = cov(Dim_i, Dim_j))$$

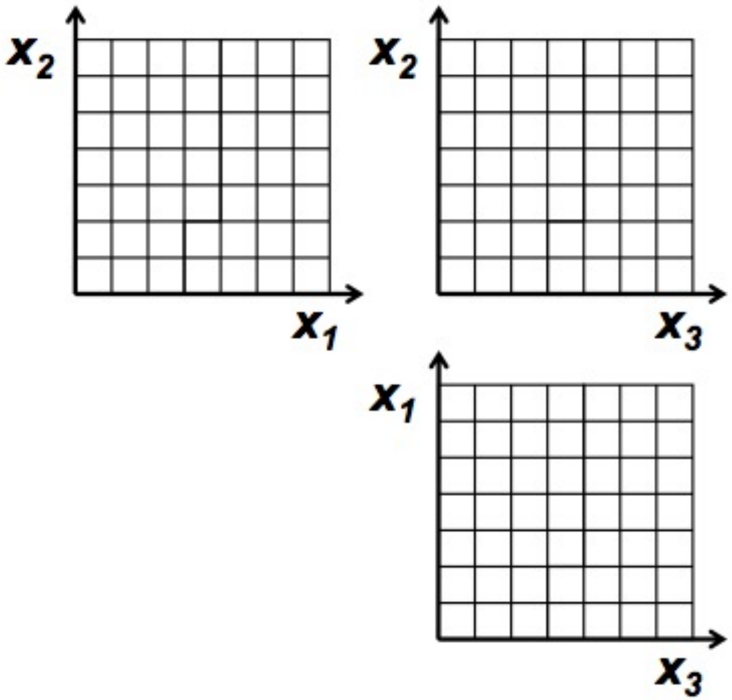
$$C = \begin{pmatrix} cov(x,x) & cov(x,y) & cov(x,z) \\ cov(y,x) & cov(y,y) & cov(y,z) \\ cov(z,x) & cov(z,y) & cov(z,z) \end{pmatrix}$$

- Values along the main diagonal describe the variance of the corresponding dimension.
- Based on its definition, it is obvious that cov(x,y)=cov(y,x), i.e., the covariance matrix is symetric with respect to its main diagonal.

Statistics

- Exercise. Obtain the covariance matrix for the given data set:

Obs.	X1	X2	X3
1	2	2	4
2	3	4	6
3	5	4	2
4	6	6	4



	x_1	x_2	x_3	$x_1-\mu_1$	$x_2-\mu_2$	$x_3-\mu_3$	$(x_1-\mu_1)^2$	$(x_2-\mu_2)^2$	$(x_3-\mu_3)^2$	$(x_1-\mu_1)(x_2-\mu_2)$	$(x_1-\mu_1)(x_3-\mu_3)$	$(x_2-\mu_2)(x_3-\mu_3)$
1												
2												
3												
4												

Eigenvectors and eigenvalues

- Consider the multiplication of a matrix by a vector:

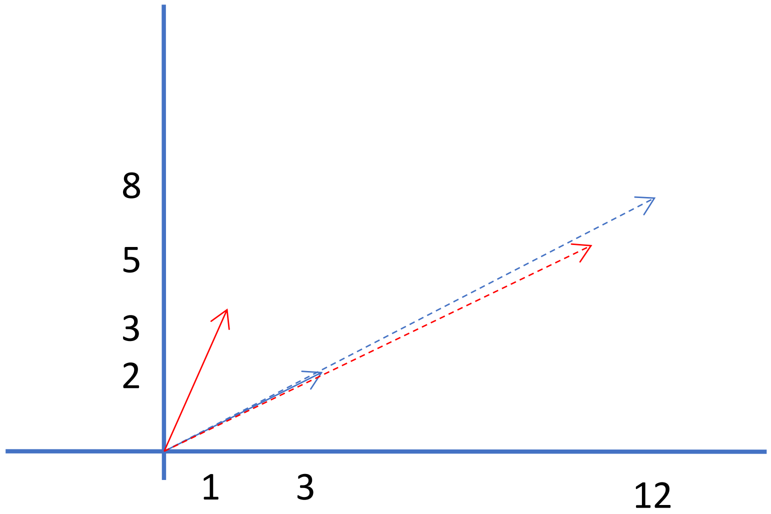
$$\begin{pmatrix} 2 & 3 \\ 2 & 1 \end{pmatrix} \times \begin{pmatrix} 1 \\ 3 \end{pmatrix} = \begin{pmatrix} 11 \\ 5 \end{pmatrix}$$

$$\begin{pmatrix} 2 & 3 \\ 2 & 1 \end{pmatrix} \times \begin{pmatrix} 3 \\ 2 \end{pmatrix} = \begin{pmatrix} 12 \\ 8 \end{pmatrix} = 4 \times \begin{pmatrix} 3 \\ 2 \end{pmatrix}$$

- In the first case, the resulting vector is not a multiple of the original vector.
- Oppositelly, in the second case, the resultant vector (12,8) is a multiple of the multiplier. As such, the latter is an **eigenvector**.
  - The correspondong **eigenvalue** is “4”

Eigenvectors and eigenvalues

- By analysing the direction of the original and resultant vectors:



- Regarding the matrix as a **transformation** (similarly to the previosuly seen transformation matrices), it can be concluded that in the second case, the direction was not changed. **This is the key insight the notion of eigenvector.**
  - The given matrix does not change the direction of its eigenvectors.

Eigenvectors and eigenvalues

- The notion of **eigenvalue** is strongly related to the **eigenvector**.
- It is the value that should by multiplied by the eigenvector to obtain the original vector.
- In the above example, 4 was the eigenvalue that corresponds to the given eigenvector.
- As such, eigenvalues and eigenvectors come in pairs and are always inter-related.

Eigenvectors and eigenvalues

- As a summary, the eigenvectors of a matrix correspond to the directions that are not changed by the transformation matrix.
- Not all matrices have eigenvectors.
- Matrices have to be square.
- A (n x n) matrix has – at most – “n” eigenvectors.
- The set of eigenvectors of a matrix (image) is widely used to describe the spatial content of that image (feature).
- In MATLAB, this eigenanalysis is made by the “eig()” function:
  - [V,D] = eig(A)
  - Returns the eigenvectors (D) and corresponding eigenvalues (V) of matrix A.

Eigenvectors and eigenvalues

- There is an important property to be stressed: the eigenvectors of a matrix are orthogonal. This is to say that they form an orthogonal basis of the matrix.
  - We are able to express every point of a data set by linear combinations of its basis-vectors.
  - **This is specially usefull for the analysis of principal components (PCA).**
  - It is usual to determine the eigenvectors/eigenvalues in their normalized version, i.e., with length normalized to 1.
  - As previously seen, the length of a vector does not affect its property of being (or not) an eigenvector.
  - Hence, having an eigenvector $(x_1, ..., x_n)$ it is usual to divide each component by the norm of this vector, in order to obtain length "1":
    - $||(x_1,..., x_n)|| = sqrt (x_1^2 + ... + x_n^2)$

Eigenvectors and eigenvalues

- **Exercise**
  - Determine, from the following vectors, which are eigenvector of the matrix given below and, if positive, determine the corresponding eigenvalue.
    - Matrix:

$$\begin{pmatrix} 3 & 0 & 1 \\ -4 & 1 & 2 \\ -6 & 0 & -2 \end{pmatrix}$$

    - Vectors:

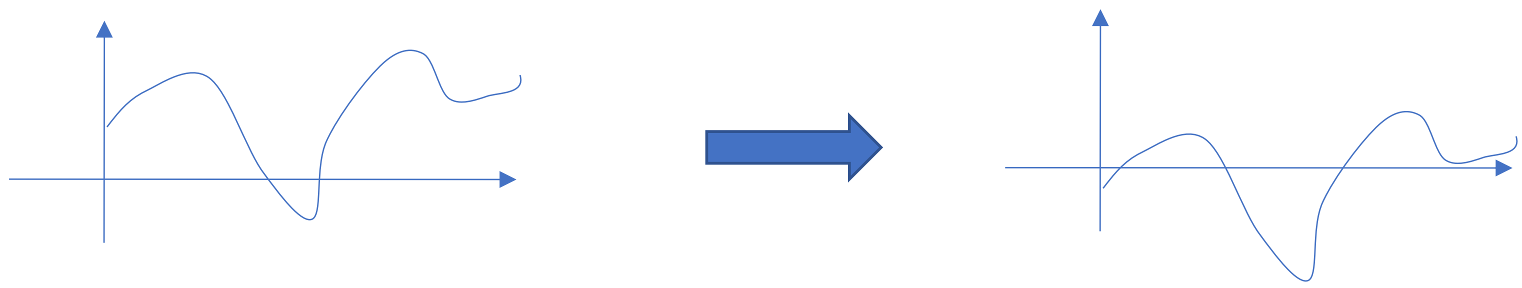
$$\begin{pmatrix} 2 \\ 2 \\ -1 \end{pmatrix} \begin{pmatrix} -1 \\ 0 \\ 2 \end{pmatrix} \begin{pmatrix} -1 \\ 1 \\ 3 \end{pmatrix} \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix} \begin{pmatrix} 3 \\ 2 \\ 1 \end{pmatrix}$$

Principal Component Analysis

- **The Principal Component Analysis (PCA)** it's a well known way to detect patterns on data, by expressing it on a way that enhances similarities or differences.
- Detecting patterns on high dimensional data is a hard task, either for humans or machines.
  - Requires huge amounts of data. An empirical rule says that at the minimum, d^2 instances are required to analyze a d-dimensional data set.
- PCA is also used to compress data (reduce dimensionality), without loosing substantial information.

Principal Component Analysis

- **Step 1.** The analysis of principal components requires a data set (with dimension n) and cardinality (k).
- **Step 2.** Removal of energy. For each dimension, the corresponding mean is subtrated to each component. As such, all dimensions of the data set have zero energy.



Principal Component Analysis

- **Step 3.** Calculus of the covariance matrix. Here, the relationships between independent components are detected, together with an assessment of the data dispersion in each dimension (by analysing the main diagonal components).
- **Step 4.** As the covariance matrix is square, it is possible to obtain the set of eigenvectors and correspºonding eigenvalues.
- **Step 4.1.** Eigenvectors normalization. All eigenvectors are normalized to have norm equal to 1.

Principal Component Analysis

- **Step 5.** Selection of components. The set of eigenvectors is sorted by decreasing order, considering the corresponding eigenvalues. From this set, the "k_1" principal components are selected.
  - This is the step that performs the reduction of dimensionality.
- **Step 6.** A transformation matrix is built, by concatenating the eigenvectors selected in the previous step.
  - This matrix will be used to represent all points in the reduced dimensionality feature space. MAT=[vect1, vect2, ... Vectk_1]

Principal Component Analysis

- **Step 7. Data Transformation. As the transformation matrix has "d" lines (corresponding to the dimension of the original feature space and** k_1 columns (corresponding to the dimension of the new feature space), when multipling each original data point by the transformation matrix, we obtain a vector of k_1 components. These are the new representation of the data points, in the principal components space.

$$\mathbf{[1 \times d] \times [d \times k_1] = [1 \times k_1]}$$

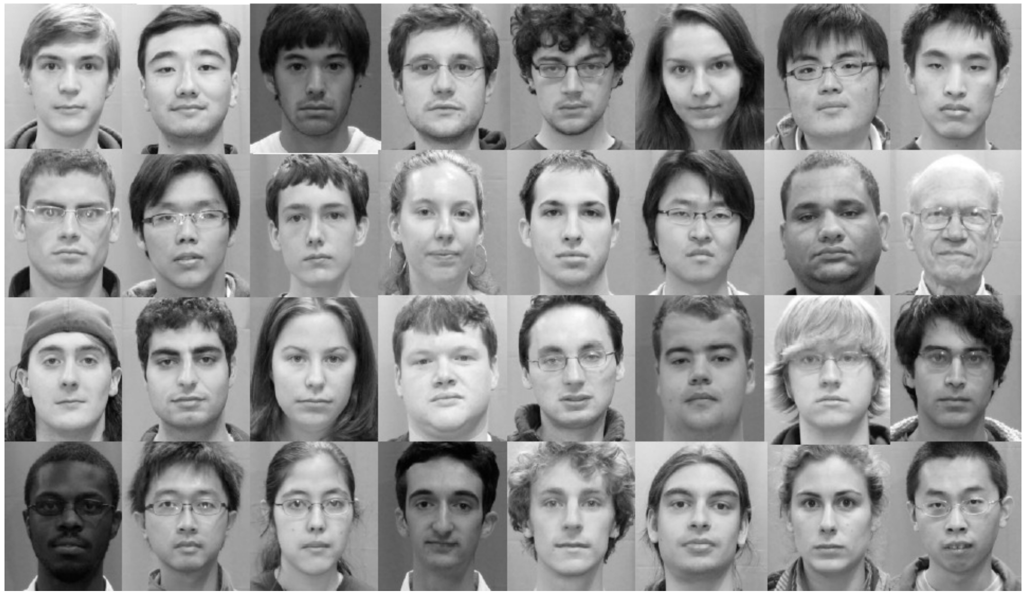
Principal Component Analysis

- **How to choose the value of k_1?**
  - The previously described process does not give any information about a strategy to select the dimensionality of the principal components feature space.
  - There is no formal rule. However, some heuristics about what is generally better exist.
- Usually, the variation in magnitude of consecutive eigenvalues (after sorting) is measured. When changes in magnitude are higher than a threshold, the selection process is stopped.
- Most frequently, the proportion of the data variability that is kept by the selected components is measured.
  - We are interested in keeping around 90-95% of the original data variability.
  - The analysis can be done by measuring the proportion of the sum of eigenvalues λ_i:
  - **Variability:**

$$= \frac{\sum_{i=1}^{M} \lambda_i}{\sum_{i=1}^{d} \lambda_i}$$

PCA: Example

- Having a set of 128 face images (with dimensions 64 x 64).
- Each face can be regarded as a point represented in a feature space of 4096 dimensions (64 x 64).

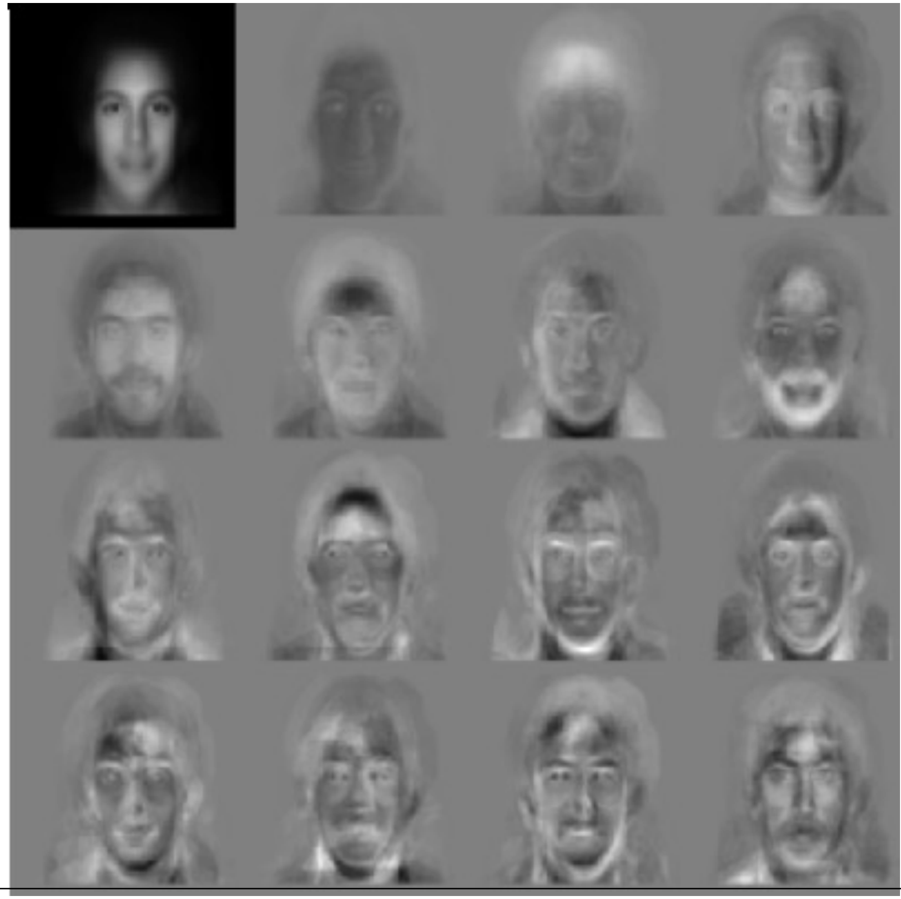


PCA: Example

- The PCA algorithm is used to select the principal components.
  - In pratice, the eigenvectors (with dimension 4096) with largest corresponding eigenvalues will be selected.
- As an example, the facial recognition process can be done in the new feature space of (much more) reduced dimension.
- Or, the PCA can be also used to represent a face, with much less information.

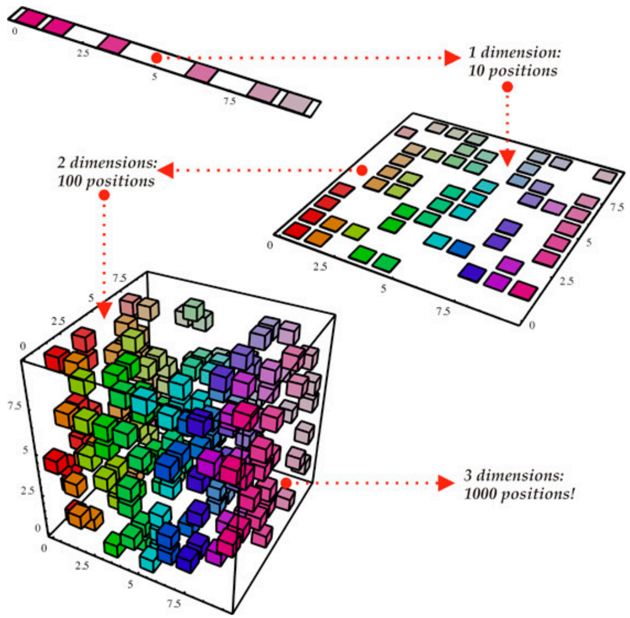
PCA: Example

- Example of the 16 principal components (eigenvectors with the largest eigenvalues) from the above data set:



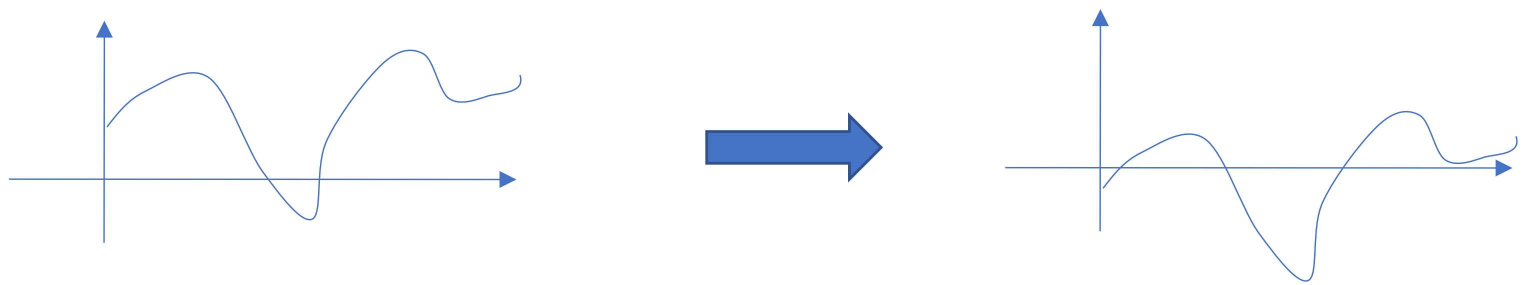
Principal Component Analysis

- **The Principal Component Analysis (PCA)** it's a well known way to detect patterns on data, by expressing it on a way that enhances similarities or differences.
- Detecting patterns on high dimensional data is a hard task, either for humans or machines.
  - Requires huge amounts of data. An empirical rule says that at the minimum, d^2 instances are required to analyze a d-dimensional data set.
- PCA is also used to compress data (reduce dimensionality), without loosing "too much" information.



Principal Component Analysis

- **Step 1.** The analysis of principal components requires a data set (with dimension n) and cardinality (k).
- **Step 2.** Removal of energy. For each dimension, the corresponding mean is subtrated to each component. As such, all dimensions of the data set have zero energy.



Principal Component Analysis

- **Step 3.** Calculus of the covariance matrix. Here, the relationships between independent components are detected, together with an assessment of the data dispersion in each dimension (by analysing the main diagonal components).
- **Step 4.** As the covariance matrix is square, it is possible to obtain the set of eigenvectors and corresponding eigenvalues.
- **Step 4.1.** Eigenvectors normalization. All eigenvectors are normalized to have norm equal to 1.

Principal Component Analysis

- **Step 5.** Selection of components. The set of eigenvectors is sorted by decreasing order, considering the corresponding eigenvalues. From this set, the "k_1" principal components are selected.
  - This is the step that performs the reduction of dimensionality.
- **Step 6.** A transformation matrix is built, by concatenating the eigenvectors selected in the previous step.
  - This matrix will be used to represent all points in the reduced dimensionality feature space. MAT=[vect1, vect2, ... Vectk_1]

Principal Component Analysis

- **Step 7. Data Transformation. As the transformation matrix has "d" lines (corresponding to the dimension of the original feature space and** k_1 columns (corresponding to the dimension of the new feature space), when multipling each original data point by the transformation matrix, we obtain a vector of k_1 components. These are the new representation of the data points, in the principal components space.

$$\mathbf{[1 \times d] \times [d \times k_1] = [1 \times k_1]}$$

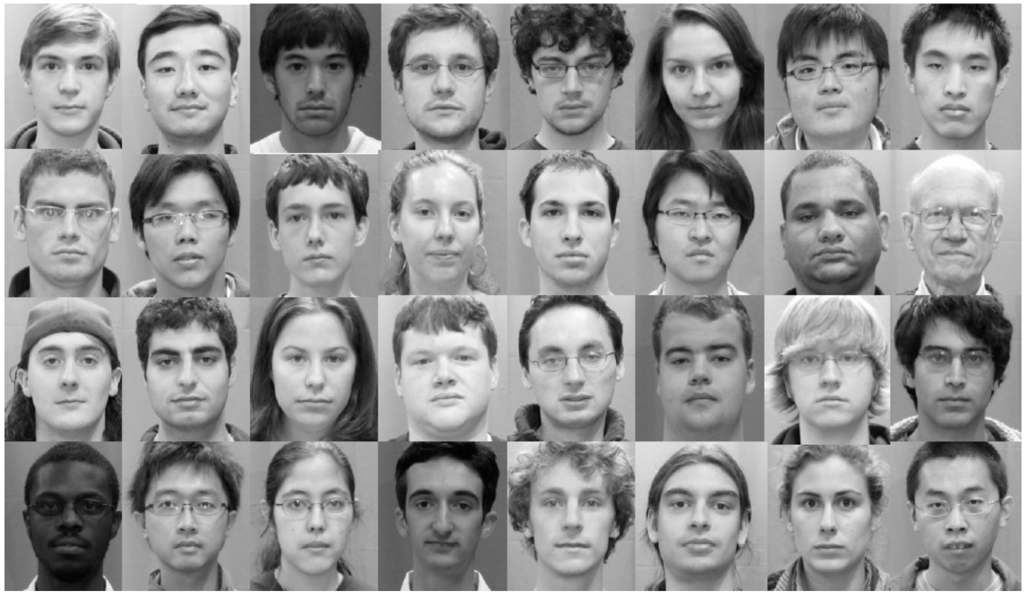
Principal Component Analysis

- **How to choose the value of k_1?**
  - The previously described process does not give any information about a strategy to select the dimensionality of the principal components feature space.
  - There is no formal rule. However, some heuristics about what is generally better exist.
- Usually, the variation in magnitude of consecutive eigenvalues (after sorting) is measured. When changes in magnitude are higher than a threshold, the selection process is stopped.
- Most frequently, the proportion of the data variability that is kept by the selected components is measured.
  - We are usually interested in keeping around 90-95% of the original data variability.
  - The analysis can be done by measuring the proportion of the sum of eigenvalues λ_i :
  - **Variability:**

$$= \frac{\sum_{i=1}^{M} \lambda_i}{\sum_{i=1}^{d} \lambda_i}$$

PCA: Example

- Having a set of 128 face images (with dimensions 64 x 64).
- Each face can be regarded as a point represented in a feature space of 4096 dimensions (=64 x 64).



PCA: Example

- The PCA algorithm is used to select the principal components.
  - In pratice, the eigenvectors (with dimension 4096) with largest corresponding eigenvalues will be selected.
- As an example, the facial recognition process can be done in the new feature space of (much more) reduced dimension.
- Or, the PCA can be also used to represent a face, with much less information.

PCA: Example

- Example of the 16 principal components (eigenvectors with the largest eigenvalues) from the above data set: